Semi-Automatic Indoor Fingerprinting Database Crowdsourcing with Continuous Movements and Social Contacts

Khuong An Nguyen
Computer Science Department
Royal Holloway, University of London
Surrey TW20 0EX, United Kingdom
Email: me@khuong.vn

Zhiyuan Luo
Computer Science Department
Royal Holloway, University of London
Surrey TW20 0EX, United Kingdom
Email: zhiyuan@cs.rhul.ac.uk

***Abstract*— Indoor localisation helps monitoring the positions of a person inside a building, without GPS coverage. In the past decade, much research effort have been invested into Indoor Fingerprinting, which is considered one of the most effective indoor tracking methods to date. In recent years, some researches started looking at crowdsourcing the fingerprinting database with the contributions from indoor users via mobile phones or laptop PCs. However, the crowdsourcing process was greatly limited due to the lack of indoor reference, in contrast to the widespread use of GPS reference for outdoor crowdsourcing. In this paper, we propose a novel idea to crowdsource the fingerprinting database without any preset infrastructure, landmarks, nor using any advanced sensors. Our idea is based on the observations that the users often carry a mobile phone with them, and there are multiple social contacts amongst those users indoor. First, we exploit the user's continuous movement indoor to refine the location prediction set. Our approach can be applied to enhance other systems. Second, we use a unique concept to detect the indoor social contacts with NFC by tapping the back of the 2 phones together. Third, we propose a novel idea to combine this social contact and the user's continuous movements to identify the exact entries with confidence in the fingerprinting database that need updating for crowdsourcing. Finally, we share our thoughts on automating the crowdsourcing process without any user input.**

I. Introduction

Indoor localisation helps monitoring the positions of a person inside a building, without GPS coverage. In the past decade, much research effort have been invested into Indoor Fingerprinting, which is considered one of the most effective indoor tracking methods to date. Fingerprinting-based approaches live above the communication layers such as WLAN, GSM, FM, and took advantage of the existing infrastructures to provide location tracking service. A few metre accuracy level has been reported in laboratory experiments. However, one of the challenges to allow widespread deployment of Fingerprinting was the maintainability of the off-line training database, which gradually becomes outdated because of environmental changes. In recent years, some researches started looking at crowdsourcing the fingerprinting database with the contributions from indoor users via mobile phones or laptop PCs. However, the crowdsourcing process was greatly limited due to the lack of indoor reference, in contrast to the successful use of GPS reference for outdoor crowdsourcing.

In this paper, we propose a novel idea to crowdsource the fingerprinting database without any preset infrastructure, landmarks, nor any advanced sensors. Our idea is based on the observations that the users often carry a mobile phone with them, and there are multiple social contacts amongst those users indoor. This information seems to be largely underused by the community so far. First, we explain how to constrain a location predictions set generated by any prediction algorithm at any position, by comparing the continuous movement data. Further, when two persons meet, their location estimations serve 2 purposes. First, it constrains the final location as an intersection of the 2 prediction sets, and removes all outliers, which are far away from their actual location, but have a similar signal readings because of signal attenuation indoor. In reality, the user often meets several other people in the same place, which greatly reduces their prediction sets at that location. The second purpose is the 'meeting place' can be used as a ground-truth reference to pinpoint the exact entries in the fingerprinting database that needs updating. Our algorithm can deliver a 'guaranteed' index with a confident level for the training data, given the prediction set may contains more than 1 location. To acknowledge when two persons meet, we propose an idea with Near Field Communication (NFC) by simply tapping the back of the 2 phones together. To our knowledge, we were the first to exploit such NFC function for the indoor localisation. Further, we explain a novel method to make our system fully automatic without much user intervention at all.

The main contributions of our paper can be addressed in four ways. First, we exploit the user's continuous movement indoor to refine the location estimation set. Our approach can be applied to enhance other current systems. Second, we propose a unique concept to detect the indoor contacts with NFC by tapping the back of the phones together. Third, we propose an idea to combine the above 2 concepts to identify the exact entries in the fingerprinting database with confidence for crowdsourcing. Lastly, we share our thoughts on automating the crowdsourcing process without user input.

II. The Future of Fingerprinting Indoor Localisation

A. Current State-of-the-Art and Challenges

Global Navigation Satellite Systems (GNSS) such as GPS have been successfully deployed in the past 2 decades, and are indispensable for outdoor navigation. However, people spend most of their times indoor, where limited or not at all GNSS service is available. The demands of daily used applications such as supermarket and hospital navigation, to emergency systems have encouraged much interest in the indoor localisation research. Fine-grained positioning systems with centimetre accuracy to coarse-grained room-level systems have been successfully reported [1], [2]. Since invented in 2001, Location Fingerprinting has gained much popularity due to its simplicity, which takes advantage of the existing building communication infrastructure such as WLAN [3]. The method has 2 phases. In the first phase, which is known as the off-line phase, a training database collects the WLAN signal at every location in the building. In the on-line phase, when a user wants to discover his position, he measures the WLAN signal at his current location, and use the previous training database to infer a closest match. Fingerprinting can be viewed as a typical classification problem, where the training database composes of examples mapping the WLAN signal (the object), to its Cartesian x, y, z co-ordinate (the label). Our task is to predict the right label for a known object. However, this prompts a question if Fingerprinting is the right direction for future indoor localisation? Below are some of our thoughts on the strengths and weaknesses of Fingerprinting.

In terms of **accuracy**, Fingerprinting is still a long way short of the extreme 3 cm achieved by those lateration and angulation-based systems [1]. Although we have seen a much improved sub-metre tracking accuracy reported in recent works, typically with the use of CSI [4]–[6], there are multiple components including the training data resolution and density, signal properties, prediction algorithm, which all contribute to the end tracking result.

Availability can be Fingerprinting's strength, thanks to the ubiquitous indoor communication infrastructure such as WLAN or Bluetooth. Other long range outdoor signals such as FM or GSM can be used to boost low coverage indoor areas. There are several reports on Fingerprinting outdoor localisation systems, both commercialised and non-commercialised, such as SkyHook[1] or OpenStreetMap[2].

Installation and ease of use have their pros and cons. In most cases, the user only needs to install an app on their mobile devices to enable tracking capability. Apart from a central server to exchange data with the users, no extra hardware is needed, because the whole idea took advantage of the existing communication infrastructure of the building. However, the initial concept of Fingerprinting does require an off-line site-survey step.

[1]http://www.skyhookwireless.com

[2]http://www.openstreetmap.org

Maintainability is the most challenging aspect of Fingerprinting. The training database becomes outdated over times, and to re-calibrate the whole tracking zone requires much labour work. This is one of the reasons Fingerprinting has yet been widely deployed in real offices.

We have not discussed other aspects such as security, risk, reliability, since they are out of the scopes for this paper. Clearly, one of the challenges for Fingerprinting to be practical is the off-line training data handling. We need a less manual labour, yet reliable concept to collect and update such database. In recent years, crowdsourcing has emerged as the front runner to tackle such issue. There are still much challenges when applying crowdsourcing into Fingerprinting, to be discussed in the next part.

B. Crowdsourcing the Fingerprinting Database

Crowdsourcing is an idea of dividing a big task into smaller sub-tasks, that can be solved separately by individuals. They can contribute to the end result at the same time, or in turn. There are several advantages for us to consider Crowdsourcing as an ideal candidate to handle the Fingerprinting database. First, many people use PC, laptops and other electronic equipments on a daily basis that are capable of receiving the WLAN signals. Further, people often carry a mobile phone with them when they are out and about. These people can be turned into mobile contributors to crowdsource the fingerprinting database un-intentionally, while tradition fingerprinting systems employed experts to pre-survey the building. There are 2 ways to crowdsource an indoor fingerprinting database, client-side or server-side. For client-side crowdsourcing, each person has an app on his mobile phone to report the latest WLAN signals at his current location to a central server. The app can run in the background without much interference to other activities. For server-side crowdsourcing, there is no custom code to be installed at the user-end at all. Instead, a custom WLAN driver is installed on the Access Points (APs) to monitor the latest WLAN signals to the registered users. At any time the user does not wish to be tracked, he simply switches off his phone's WLAN adapter, in the case of server-side tracking, or exits the app for client-side tracking.

One of the major challenges for indoor fingerprinting crowdsourcing is the lack of 'ground-truth' references between the contributors' data and the training data. Outdoor crowdsourcing systems relied on GPS to provide such reference. In the last few years, some research effort has been spent to tackle this issue, such as providing a graphic user interface for the users to manually input their current location [7], [8], or deploying fixed landmarks throughout the tracking zone so that the users make contributions at specific positions in the building [9]. The most notable work is the use of inertial mobile phone sensors (accelerometer, compass, gyroscope), combining with a site map to provide location reference [10]. In our work, we avoid using extra infrastructure which are not practical to deploy, nor advanced sensors which are noisy indoor. Our ideas exploit the indoor social contact aspects and NFC to crowdsource the Fingerprinting database, which we will discuss soon.

C. Related Work

There are several Pedestrian Dead-Reckoning (PDR) based systems, that use inertial sensors (accelerometer, compass, gyroscope) in the mobile phones to navigate around a building [10]–[13]. Since these systems are independent of the WLAN signal, they use their own navigation capability to provide location reference for the WLAN signals collected while the user navigates the building. These PDR systems are closest to infrastructure-less automatic crowdsourcing systems. The challenges for those systems are that the sensors in current smart phones were mainly included for basic app support, rather than for robust tracking purpose, therefore they are susceptible to indoor noise. Compass sensor does not work at all in many offices. Further, the position the user holding the phones and its orientation affect the accelerometer readings. Traditional PDR systems attach a small device on the users' feet to measure the stride and step length, while it is not feasible to stick a smart phone onto the users' feet. Our work avoid using such inertial sensors.

There have been several NFC-based indoor positioning systems. In [7], multiple QR tags are set up in fixed locations in the building. The users scan the tags with the phone's camera to reveal their current location to the system. In [14], multiple RFID tags are deployed in a similar manner to the QR codes. These RFID tags, however, enable automatic signal collection when a user passes by. We avoid using such tags, since extra infrastructure (QR codes, RFID tags) must be set up for a new building beforehand. Other systems require manual inputs from the users via a GUI to identify which position the latest WLAN signals come from [7]. However, the users can only recognise their current locations by room number, resulting in coarse-grained tracking level. We would prefer the crowdsourcing process to be executed with less user intervention, or not at all if possible.

III. Exploiting Continuous Movements and Social Contacts For Fingerprinting Crowdsourcing

A. Indoor Fingerprinting Assumptions

Our ideas rely on the following 2 assumptions. Through-out the paper, we will refer back to them for further clarifications.

1) The quality of the training database decreases gradually, but not instantly. The dynamic environment (furniture re-arrangement, human movements, humidity) contributes to the changes of WLAN signals in the building.
2) The user cannot jump a long distance in a short period of time. Typically, it is unlikely the user can travel more than 5 metres within 3 seconds by walking.

B. Extra Information From Continuous Movements Indoor

When Alice wants to navigate the building, she opens the tracking app on her mobile phone (in the case of client-side tracking), or simply switches on the WLAN adapter to let the APs recognise her phone (in the case of AP-side tracking), as discussed previously. The system measures the WLAN signal strength (RSSI) between Alice and the nearby APs, and calculates a set of location estimations $y_A = \{a_1, a_2, \dots, a_N\}$ with $a_i = (x, y, z)$ is the Cartesian co-ordinate vector, in which Alice may currently resides. While previous solutions combined these locations or prioritised certain prediction, we will treat all these predictions equally for now. In our experience, not any of these locations is wrongly predicted by the algorithms. In fact, the areas around Alice have a similar WLAN reading because of the indoor signal attenuation, a typical challenge of indoor localisation. Also, the signal at her current location recorded in the database may have already changed since the last time it was collected.

A moment later, Alice moves away. The system measures the RSSI from her new location, and another independent set of prediction locations is returned $y'_A = \{a'_1, a'_2, \dots, a'_M\}$. Based on the second assumption (Section III.A) that Alice cannot jump a long distance in a short period of time, we know that these 2 prediction sets are in close proximity. Therefore, by comparing the Cartesian distance between those sets, we can select the top current predictions that are likely to be reached from all preceding predicted locations. First, we remove the isolated outliers in y'_A, that are 5 metres or more to all predictions in y_A in the Cartesian space. Second, we calculate the distance between each prediction $a'_i \in y'_A$ to the whole preceding location set y_A, and retain the top 50% predictions in y_A with the smallest distances. When Alice moves to another new location, we repeat this process again. However, this time we disregard the original prediction set y_A, and only consider y'_A to refine her current prediction location.

In summary, by considering the indoor continuous movements, we were able to accumulate the location prediction history as the user navigates the building. Our approach refines the current location set by removing the violated predictions, based on the most recent location's predictions. In the next section, we introduce the indoor social contacts idea to crowdsource a fingerprinting database.

C. Exploiting the Indoor Social Contacts for Crowdsourcing

We use the above scenario with Alice navigating the building. At some moment, Bob, who is using the same tracking system, happens to walk by. The system is also keeping track of Bob's independent prediction location history. If we can acknowledge that Bob & Alice are in the same position, their current location estimations can be further reduced to the intersection of the 2 prediction sets. In daily environment, more than often many people are in the same location at different times through out the day. For example, if Carol happens to walk by the position Bob & Alice are currently in, their locations are greatly constrained to much fewer overlapped predictions. Regardless of what prediction algorithm we use, if the final intersected set contains only 1 prediction, we are certain that this is the training database entry to be updated with the latest WLAN signals. In reality, the final set often contains more than 1 prediction, due to the similarity of the indoor signals in a small area. We propose a novel algorithm, based on our previous work on Conformal Prediction Indoor Localisation, to reduce the size of the prediction sets with each user's confidence level [15], [16]. Our algorithm is summarised as follows.

Giving a training database $B = (z_1, z_2, \ldots, z_{n-1})$ mapping Cartesian co-ordinate (the label set $\mathbf{Y}$) to WLAN signals (the object set $\mathbf{X}$), a WLAN reading at an unknown location (a new object z_n), and a pre-defined confidence level, our algorithm selects a set of examples in the database to match this new sample. We treat Fingerprinting as a classification problem, because our label set is finite. Each example z_i is a combination of a WLAN vector $RSSI_i = (s_1^i, s_2^i, \ldots, s_n^i)$ and the Cartesian co-ordinate $L_i = (d_i^x, d_i^y, d_i^z)$. To evaluate the difference amongst the examples, we employed the 'Weighted K-nearest neighbours' (W-KNN) as the underlying algorithm to compute the 'nonconformity score' α. We assume the correct position to be each of every recorded locations in $\mathbf{Y}$. A prediction region of K examples is $R^{\varepsilon}(z_1, z_2, \ldots, z_K) \subset \mathbf{Y}$. To calculate the similarity between 2 WLAN distributions P_X and P_Y, we use the symmetrised Kullback-Leibler formula, with M is the number of bins in the histogram, and N is the number of APs.

$$Sym\_D_{KL}(P_X, P_Y) = D_{KL}(P_X||P_Y) + D_{KL}(P_Y||P_X) \quad (1)$$

where

$$D_{KL}(P_X \mid\mid P_Y) = \sum_{j=1}^{N}\sum_{i=1}^{M} P_X^j[i] \; log_2 \frac{P_X^j[i]}{P_Y^j[i]} \quad (2)$$

With the above equation, we find K examples $(z_1, \ldots, z_K)$ in training database B with the smallest difference $D_{KL}(P_i, P_u)$ to the new sample U, and having the same label $L_i = (d_i^x, d_i^y, d_i^z)$ with $U(1 \le i \le K)$. We then calculate a weighted average location $e_{sm} = (d_{sm}^x, d_{sm}^y, d_{sm}^z)$ from these K examples (ϵ is a small constant to prevent division by zero).

$$d_s^{x,y,z} = \frac{\displaystyle\sum_{i=1}^{K} \frac{1}{D_{KL}(P_i, P_u) + \epsilon} \;\; d_i^{x,y,z}}{\displaystyle\sum_{i=1}^{K} \frac{1}{D_{KL}(P_i, P_u) + \epsilon}} \;. \quad (3)$$

Similarly, we find another K entries $(z'_1, \ldots, z'_K)$ in training database B with smallest distances $D_{KL}(P_i, P_u)$ to the new sample U, this time with a different label $L_i = (d_i'^x, d_i'^y, d_i'^z)$ to $U(1 \le i \le K)$. Another weighted average location $e_{df} = (d_{df}^x, d_{df}^y, d_{df}^z)$ is calculated from these K entries. Our nonconformity measure is calculated as

$$\alpha = \sqrt{(d_{sm}^x - d_{df}^x)^2 + (d_{sm}^y - d_{df}^y)^2 + (d_{sm}^z - d_{df}^z)^2} \quad (4)$$

With the above equation, we calculate the nonconformity score α_i, with $i = 1, \ldots, l$, for every example in the database B. The p-value for a possible label $\hat{y}$ is calculated as

$$p(\hat{y}) = \frac{\#\{i = 1, \ldots, l+1 : \alpha_i \ge \alpha_{l+1}\}}{l+1} \;. \quad (5)$$

Given a significance level ε beforehand, the assumed label is accepted as a correct label for the new sample, if and only if p-value $> \varepsilon$. All accepted locations form a prediction region, which guarantees to contain the correct position along with the associated confidence level. A proof of our algorithm and more details can be found in our previous work in [15], [16].

There are 2 options to reduce the size of the prediction set with our algorithm. First, we can manually decrease the confidence level of each person. Second, we can pro-actively pick the top predictions with the biggest p-values only (the top 50% predictions for example). Ideally, we prefer a high confidence level while maintaining a minimal size of prediction set. Our approach was one of the first to provide such confidence level for each prediction. In the experiment section, we will evaluate the trade-off between confidence level and prediction size.

An important requirement to implement our idea is that we must reliably and correctly detect that Bob & Alice are in the same position. The simplest solution is to let the users indicate when such contact has happened themselves. The limitation in previous works, where the users manually input their current positions via a GUI, or scan the tags deployed beforehand in the building, is that the location indicated by the user may not match the collected WLAN signals. This is due to the varying distances between the phone's camera and the tags, or because the users do not input their current locations correctly. Our idea overcame these limitations by using Near Field Communication (NFC) to detect the phone's contact correctly without extra landmarks. Since early 2011, smart phones were equipped with NFC chip, which allows them to establish close proximity connection to another phone within a few inches. In many Android phones, these chips are located at the back of the device. For our purpose, we just want a confirmation that the 2 persons are in the same location, and by tapping the back of their phones together, we have a simple, yet accurate solution. Since NFC between 2 phones only work within a few inches, the system can indicate precisely when the 2 phones are tapped, then collects the latest WLAN signals at that moment. To our knowledge, we were the first to utilise such function for the indoor localisation research.

In summary, our idea provides a simple and effective solution to detect an indoor contact by tapping the back of the phones together. This is our 'ground-truth' reference to combine the prediction sets of multiple persons in the same location. Further, we associate a confidence level for each user to reduce the size of their prediction sets. The overlapped predictions from multiple users pinpoint the correct entries in the training database for crowdsourcing.

D. Bringing It All Together

Figure 1 demonstrates the progress of our crowdsourcing scheme. The system first returns a prediction set for Alice's initial unknown location. As she navigates the building, the app periodically measures the current signal strength to refine Alice's location estimations, based on her preceding location's prediction. At any moment, Bob is detected via NFC. This is a ground-truth reference signalling that Bob and Alice's prediction sets are overlapped, and the intersected predictions are candidates for crowdsourcing. Since Bob and Alice have their own prediction location history, their prediction sets are different. By adjusting their own confidence levels, the system further reduces the size of the prediction sets.

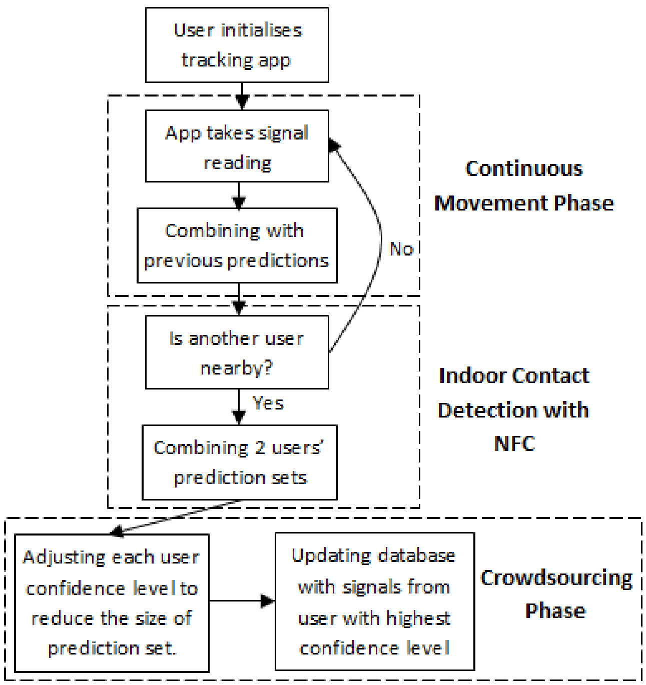


Fig. 1. Crowdsourcing Steps

IV. Empirical Experiments

A. Test beds

We used two test beds collected in real offices. Both test beds are divided into squared grids. The first test bed (TB 1) has a dense 30cm resolution, while the second one (TB 2) has a sparser 1.5m resolution [15]. For simplicity, all users possess the same mobile device in both training and real-time phases.

B. Evaluations

Figure 2 demonstrates the effectiveness of our ideas in reducing the size of the prediction set at any moment, by monitoring the user's continuous movements. In the example, we managed to remove 40% percents of predictions while keeping the correct one. It is also worth noting that the area of interest formed by the remaining predictions (the circled predictions) is tighter with our approach.

Next, we evaluate our indoor contact detection via NFC idea. If we only use the 2 location prediction sets collected at the moment the 2 users tap their phones, the averaged number of overlapped predictions is above 10 for the first test bed, and is around 5 for the second test bed. This overlapped portion occupies 70% to 85% of the whole prediction set, for both data sets. Such high proportion of similar predictions are expected, because the 2 signal sets are collected in the same position. We would not expect them to be 100% similar because of the orientation of the phones, and the way user holds the phones. However, as we combine this prediction

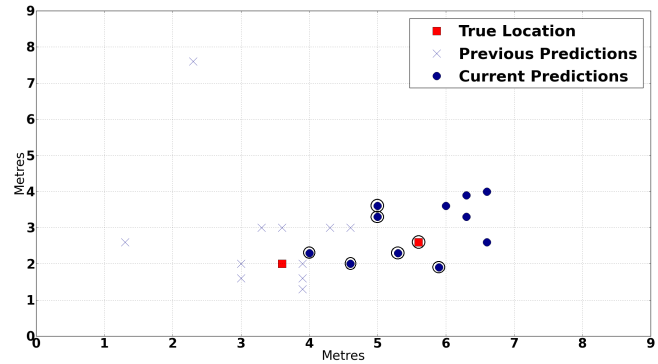


Fig. 2. Reduced Prediction Region From Continuous Tracking (TB 1)

set with the one previously generated from our continuous movement tracking scheme, the intersected portion becomes much smaller (the circled predictions in Figure 3). Although the users are currently in the same spot, they had their own navigation history, which helps removing certain prediction that they are not likely to reach from their previous locations. In the example, Alice & Bob's current location predictions are reduced to the intersected portion of the 2 circles, which contains just 2 predictions including the correct one.

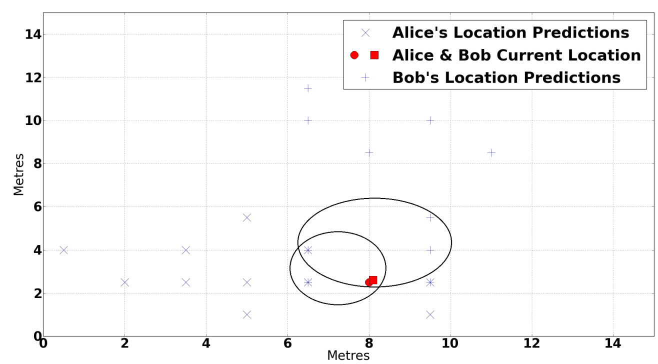


Fig. 3. Prediction Region From Both Combining Continuous Movement & Indoor Contact (TB 2)

So far, we have not discussed the confidence level parameter, which was preset at 95% for all above experiments. Figure 4 demonstrates the amount of predictions removed by decreasing this confidence level. Overall, it is safe to reduce our confidence level to 70% and 75% for the first and second test bed respectively, without losing the correct prediction. By doing so, we managed to remove up to 30% of predictions.

Overall, by averaging the intersected predictions from 2 users, we achieved less than 1.5 metres error, with 80% confidence (Figure 5). Our system can achieve near maximum database resolution accuracy, although it is not quite fair to compare ours with other existing systems, because such accuracy is obtained when an indoor contact with other users happens, and our purpose is to crowdsource the database, rather than providing location tracking.

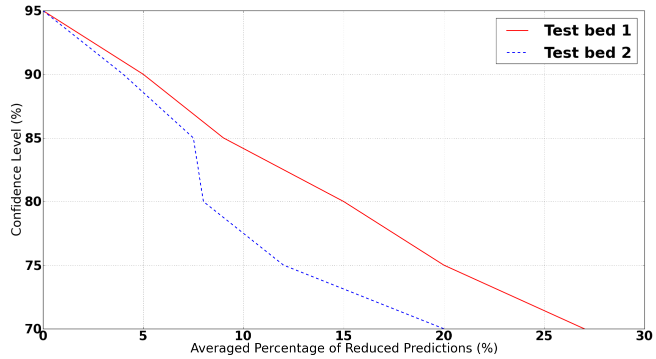
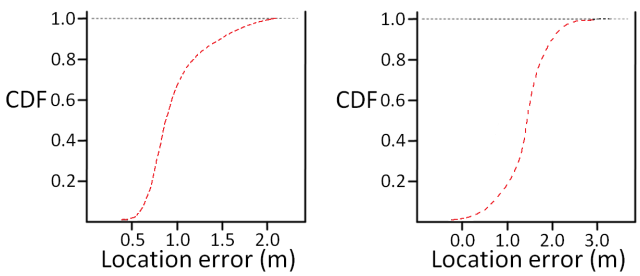


Fig. 4. Adjusting the Confidence Levels



(a) Test bed 1

(b) Test bed 2

Fig. 5. Performance Accuracy

V. Conclusions

A. Main Contributions

We have proposed a novel idea to crowdsource the fingerprinting database without any preset infrastructure, landmarks, nor any advanced sensors. Our ideas base on the observations that the users often carry a mobile phone with them, and there are multiple indoor contacts amongst those users. This information seems to be largely underused by the community so far. First, we exploited the user's continuous movement to refine the location estimation set by removing the outliers. Our approach is generic and can be applied to other current systems. We then proposed a unique concept to correctly detect the indoor contacts with NFC by tapping the back of the phones together. Finally, we define a confidence level for each user's prediction set, which can be adjusted to reduce the size of the set.

B. Future Work

Ideally, we prefer a fully automatic crowdsourcing system, where the fingerprinting database is automatically updated with the latest WLAN signals from the contributors, without extra infrastructure, nor any user intervention. One might assumes that when 2 persons are in the same position, they should observe the same wireless signals from nearby APs, therefore, their contact can be detected off-line by analysing the signals. However, this assumption does not strictly hold for both indoor and outdoor. In our other work, we calculate a 'matching rate' value, based on the APs appearance to work out the possibility that 2 users are in the same location. Our experiments showed that the average matching rate was around 60%, even when they were in the same position. This indication serves little purpose when the users are a few metres apart. However, we observed that the matching rate does reach 100% when the 2 phones are not moving at all, which might be applicable for crowdsourcing, since the users often stand still to talk to other people nearby. Further, although our initial approach does not require a site map of the building at all, such map can be combined with our continuous movement approach to remove the violated predictions, such as wall penetration.

Acknowledgment

The authors would like to thank the Computer Science department of Royal Holloway for the partial funding of this research. Khuong Nguyen would like to thank the CPHUD of Danang city for supporting his work.

References

[1] R. Want, A. Hopper, V. Falcao, and J. Gibbons, "The active badge location system," *ACM Transactions on Information Systems (TOIS)*, vol. 10, no. 1, pp. 91–102, 1992.

[2] M. Youssef and A. Agrawala, "The horus wlan location determination system," in *Proceedings of the 3rd international conference on Mobile systems, applications, and services*. ACM, 2005, pp. 205–218.

[3] P. Bahl and V. N. Padmanabhan, "Radar: An in-building rf-based user location and tracking system," in *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 2. Ieee, 2000, pp. 775–784.

[4] Z. Yang, Z. Zhou, and Y. Liu, "From rssi to csi: Indoor localization via channel response," *ACM Computing Surveys (CSUR)*, vol. 46, no. 2, p. 25, 2013.

[5] R. Crepaldi, J. Lee, R. Etkin, S.-J. Lee, and R. Kravets, "Csi-sf: Estimating wireless channel state using csi sampling & fusion," in *INFOCOM, 2012 Proceedings IEEE*. IEEE, 2012, pp. 154–162.

[6] S. Sen, R. R. Choudhury, B. Radunovic, and T. Minka, "Precise indoor localization using phy layer information," in *Proceedings of the 10th ACM Workshop on Hot Topics in Networks*. ACM, 2011, p. 18.

[7] J.-g. Park, "Indoor localization using place and motion signatures," Ph.D. dissertation, Massachusetts Institute of Technology, 2013.

[8] J. Ledlie, J.-g. Park, D. Curtis, A. Cavalcante, L. Camara, A. Costa, and R. Vieira, "Molé: A scalable, user-generated wifi positioning engine," *Journal of Location Based Services*, vol. 6, no. 2, pp. 55–80, 2012.

[9] S. Chaudhry, "Indoor location estimation using an nfc-based crowdsourcing approach for bootstrapping," 2013.

[10] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen, "Zee: zero-effort crowdsourcing for indoor localization," in *Proceedings of the 18th annual international conference on Mobile computing and networking*. ACM, 2012, pp. 293–304.

[11] H. Wang, S. Sen, A. Elgohary, M. Farid, M. Youssef, and R. R. Choudhury, "No need to war-drive: unsupervised indoor localization," in *Proceedings of the 10th international conference on Mobile systems, applications, and services*. ACM, 2012, pp. 197–210.

[12] M. Azizyan, I. Constandache, and R. Roy Choudhury, "Surroundsense: mobile phone localization via ambience fingerprinting," in *Proceedings of the 15th annual international conference on Mobile computing and networking*. ACM, 2009, pp. 261–272.

[13] I. Constandache, R. R. Choudhury, and I. Rhee, "Towards mobile phone localization without war-driving," in *INFOCOM, 2010 Proceedings IEEE*. IEEE, 2010, pp. 1–9.

[14] M. Lee and D. Han, "Qrloc: User-involved calibration using quick response codes for wi-fi based indoor localization," in *Computing and Convergence Technology (ICCCT), 2012 7th International Conference on*. IEEE, 2012, pp. 1460–1465.

[15] K. Nguyen and Z. Luo, "Conformal prediction for indoor localisation with fingerprinting method," in *Artificial Intelligence Applications and Innovations*. Springer, 2012, pp. 214–223.

[16] V. Vovk, A. Gammerman, and G. Shafer, *Algorithmic learning in a random world*. Springer, 2005.